

oplaydo1 (Linux)

- In 'Fork and Build' we built the 'DR_pi' plugin.
- In Part 1 of this guide the 'DR_pi' plugin is renamed 'oplaydo1_pi'. The [icon is changed](#).
- In [Part 2](#) the dialog is changed to allow the user to input 'Start' and 'Finish' positions and generate a GPX file for displaying a route in OpenCPN. - In 'Fork and Build' we made the 'DR_pi' plugin.

Part 1: Rename the plugin and make a new icon

Additional requirements

GIMP (Free & Open Source Image Editor). Install this using the 'Ubuntu Software Centre' (Search for GIMP)

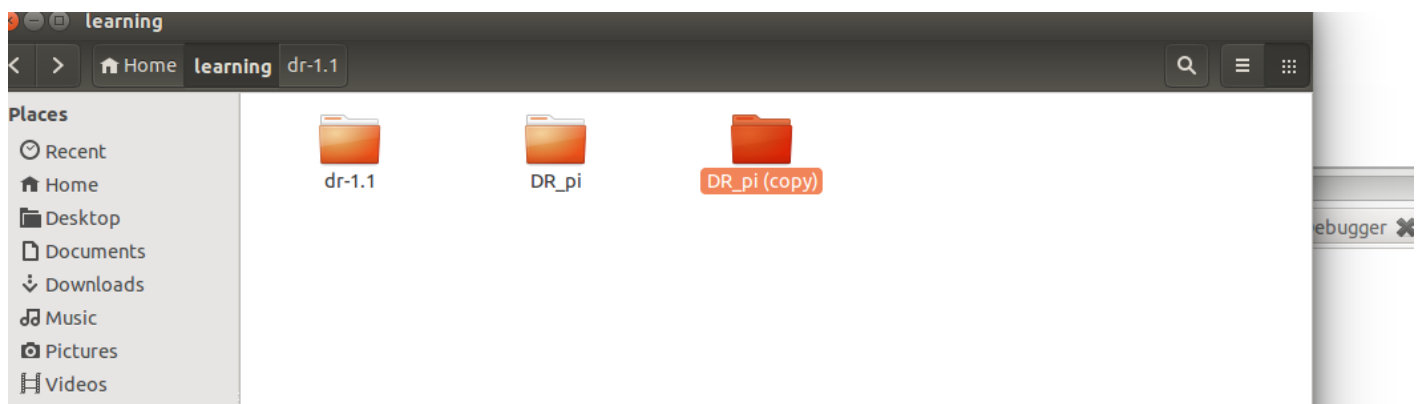
Perl (Check it is installed by using the 'Ubuntu Software Centre')

[png2wx.pl \(Perl Script\)](#)

How these programs/script are used will be explained as the guide progresses.

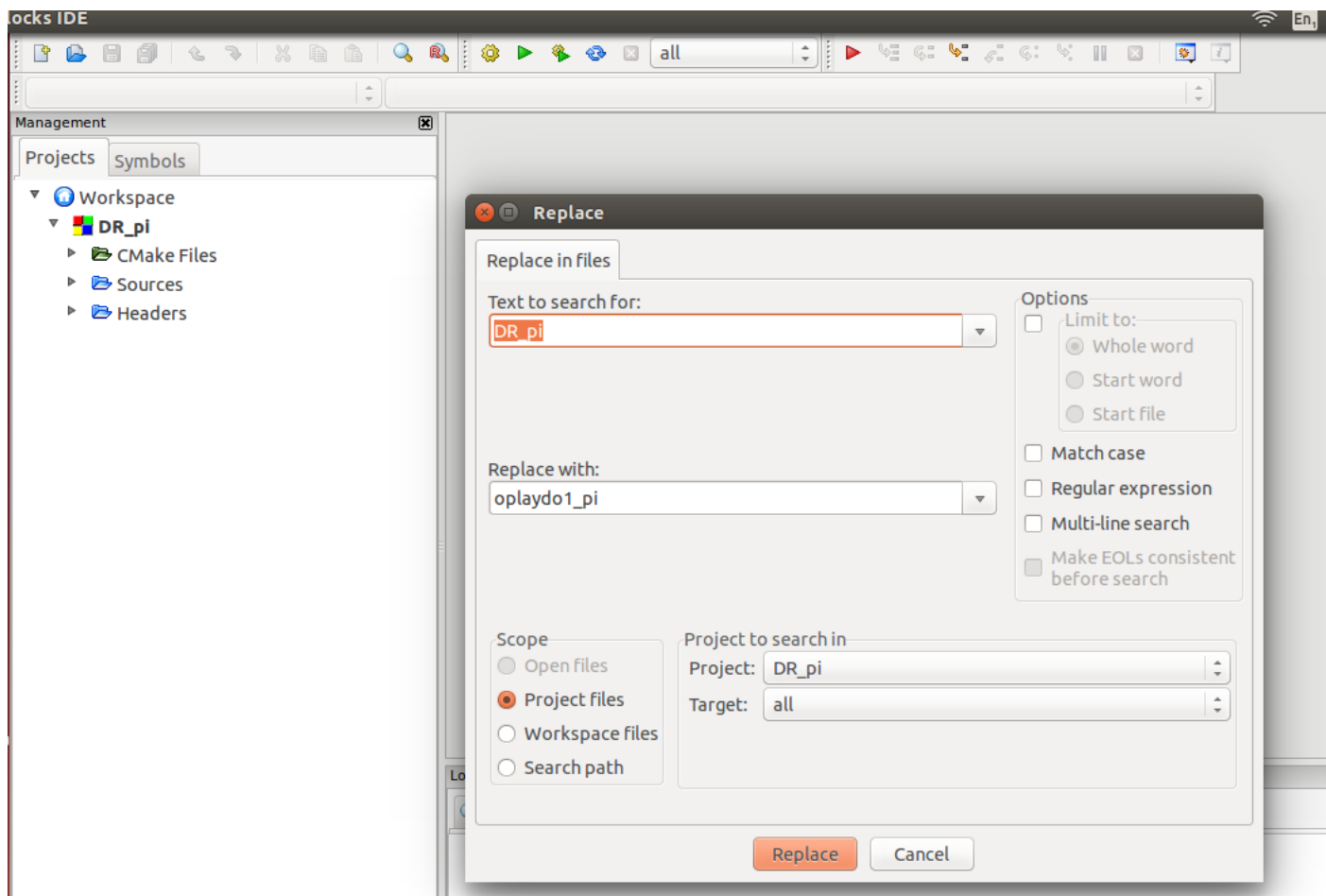
Rename and rebuild

- In 'Fork and Build' we made the 'DR_pi' plugin.
- In this section 'DR_pi' is edited and a new plugin 'oplaydo1_pi' is made from 'DR_pi'.
- Before starting it may be worth making a copy of 'DR_pi'.

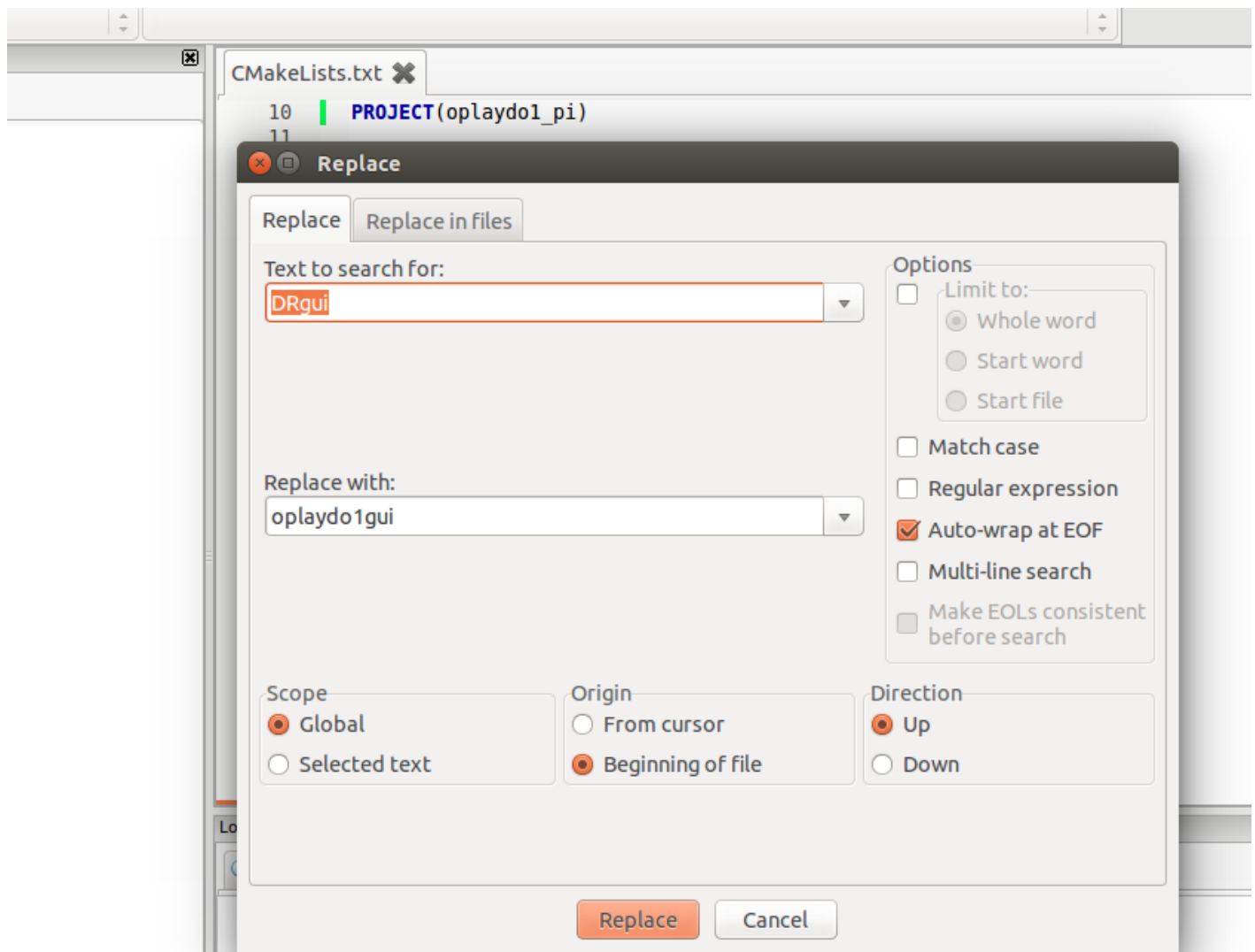


- We use CodeBlocks as the editor.

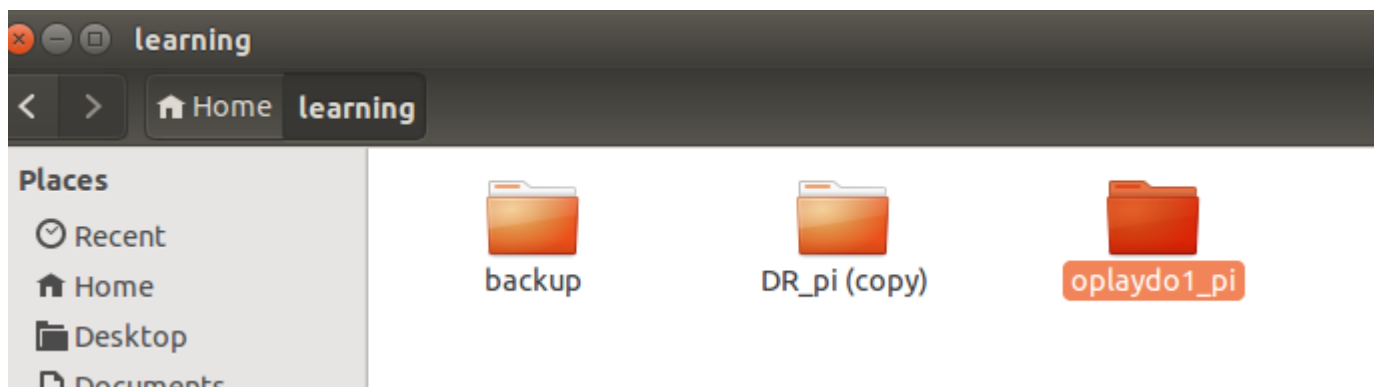
- Open the 'DR_pi.cbp' CodeBlock project with 'CodeBlocks'.
- Replace 'DR_pi' with 'oplaydo1_pi'.



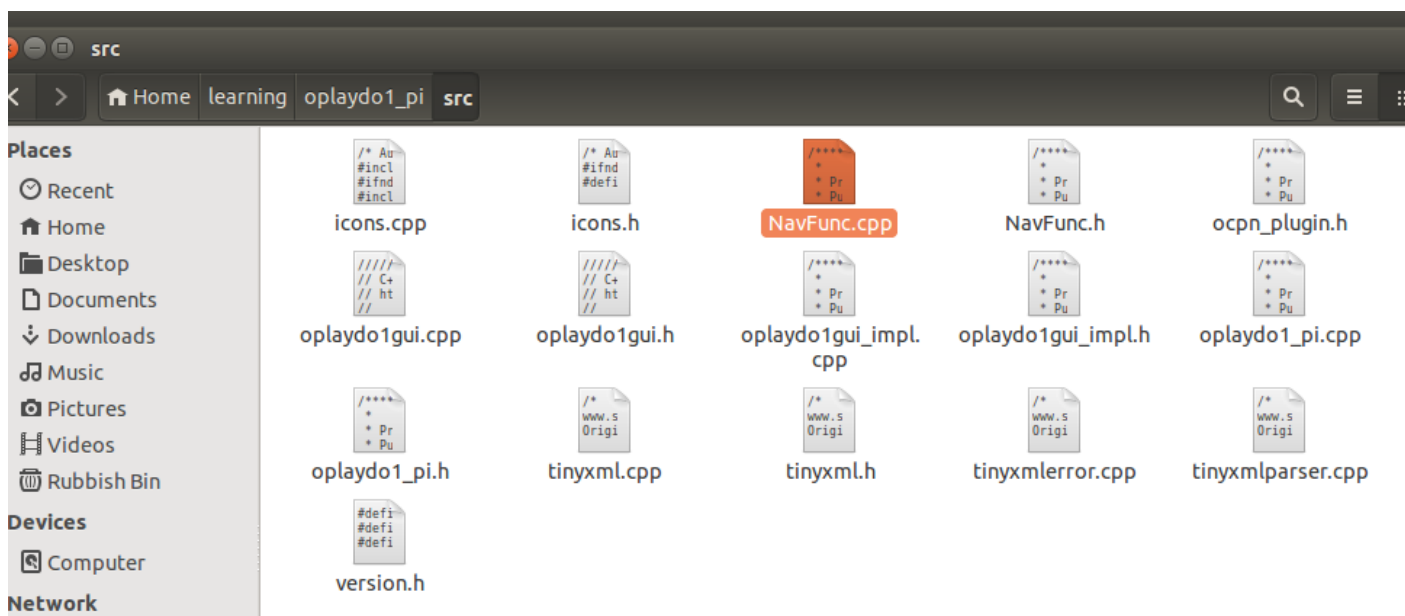
- Open the file 'CMakeList.txt' file in 'DR_pi' and edit the file names. Change 'DRgui' to 'oplaydo1gui'.
- Change all the 'DR' to 'oplaydo1'.



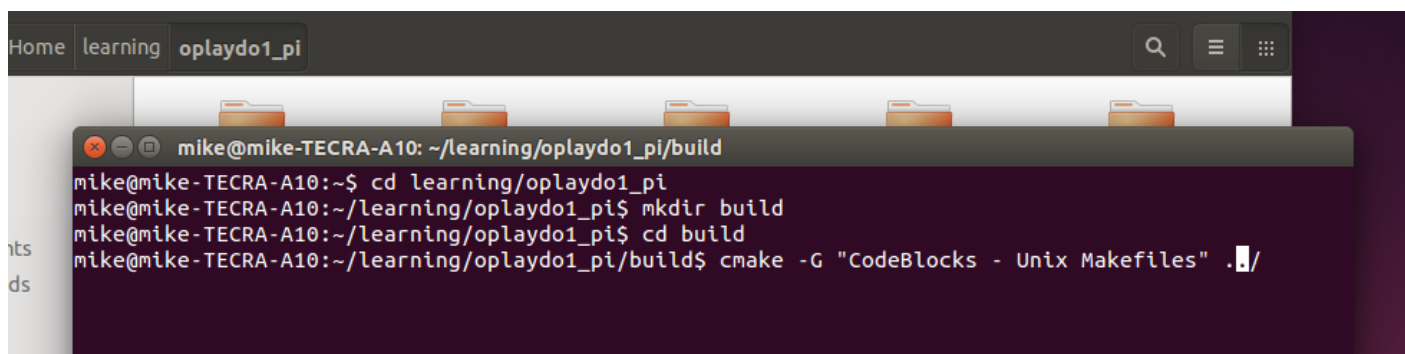
- Save your work.
- The old 'DR_pi' folder is renamed 'oplaydo1_pi'.



- In the 'oplaydo1_pi' folder rename files, changing the 'DRgui..filename' to 'oplaydo1gui..filename'.
- The files 'NavFunc.cpp' and 'NavFunc.h' can be removed because these functions will not be used.



- Delete the 'old' 'build' folder. Make a new 'build' folder for the project.
- Make a new 'CodeBlocks' project called 'oplaydo1_pi' using 'Terminal'.



```

mike@mike-TECRA-A10: ~/learning/oplaydo1_pi/build
-- Found X11: /usr/lib/i386-linux-gnu/libX11.so
-- Found OpenGL: /usr/lib/i386-linux-gnu/libGL.so
-- Found OpenGL...
--   Lib: /usr/lib/i386-linux-gnu/libGLU.so/usr/lib/i386-linux-gnu/libGL.so/usr/lib/i386-linux-gn
u/libSM.so/usr/lib/i386-linux-gnu/libICE.so/usr/lib/i386-linux-gnu/libX11.so/usr/lib/i386-linux-gnu
/libXext.so
--   Include: /usr/include
-- Found Gettext: /usr/bin/msgmerge (found version "0.18.3")
-- Found BZip2: /usr/lib/i386-linux-gnu/libbz2.so (found version "1.0.6")
-- Looking for BZ2_bzCompressInit in /usr/lib/i386-linux-gnu/libbz2.so
-- Looking for BZ2_bzCompressInit in /usr/lib/i386-linux-gnu/libbz2.so - found
-- Found ZLIB: /usr/lib/i386-linux-gnu/libz.so (found version "1.2.8")
-- Configuring done
CMake Error at CMakeLists.txt:47 (ADD_LIBRARY):
  Cannot find source file:

    src/NavFunc.h

Tried extensions .c .C .c++ .cc .cpp .cxx .m .M .mm .h .hh .h++ .hm .hpp
.hxx .in .txx

-- Build files have been written to: /home/mike/learning/oplaydo1_pi/build
mike@mike-TECRA-A10:~/learning/oplaydo1_pi/build$

```

- Using 'CodeBlocks' open the new project 'oplaydo1_pi.cbp'.
- Build the new project.
- Some errors will occur, where 'DR' needs to be changed to 'oplaydo1'.
- The reference to the include file 'NavFunc.h' is removed, as that file has been deleted. This will have caused other problems due to missing functions and where a function is no longer available it is commented out.

Calculate(wxCommandEvent& event, bool write_file, int Pattern) : void

src/oplaydo1gui_impl.h

src/oplaydo1_pi.cpp

src/oplaydo1gui.cpp

src/oplaydo1gui_impl.cpp

```

354 //DistanceBearingMercator(latN[i + 1], lonN[i + 1], latF, lonF, speed);
355
356 if (myDistForBrng > speed){
357
358     // put in the DR positions
359     //
360     count_pts = (int)floor(myDistForBrng/speed);
361     //
362     remaining_dist = myDistForBrng - (count_pts*speed);
363     //DistanceBearingMercator(latN[i + 1], lonN[i + 1], latF, lonF, speed);
364
365     for (c = 1; c <= count_pts ; c++){

```

Logs & others

Search results

Build log

Build messages

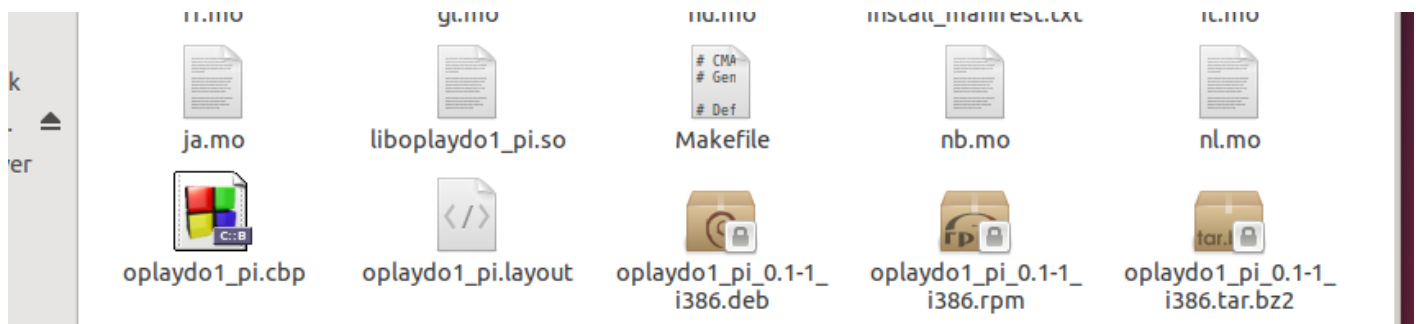
Debugger

File	Line	Message
=== Build: all in oplaydo1_pi (compiler: GNU GCC Compiler) ===		
/home/mike/learn...		In member function 'bool Dlg::OpenXML()':
/home/mike/learn...	112	warning: suggest parentheses around assignment used as truth value [-Wparentheses]
/home/mike/learn...	94	warning: unused variable 'my_count' [-Wunused-variable]
/home/mike/learn...		In member function 'void Dlg::Calculate(wxCommandEvent&, bool, int)':
/home/mike/learn...	272	warning: variable 'latF' set but not used [-Wunused-but-set-variable]
/home/mike/learn...	272	warning: variable 'lonF' set but not used [-Wunused-but-set-variable]
/home/mike/learn...	294	warning: unused variable 'routepts' [-Wunused-variable]
/home/mike/learn...	297	warning: unused variable 'myBrng' [-Wunused-variable]
/home/mike/learn...	299	warning: variable 'remaining_dist' set but not used [-Wunused-but-set-variable]
/home/mike/learn...	299	warning: variable 'myLast' set but not used [-Wunused-but-set-variable]
/home/mike/learn...	306	warning: unused variable 'skip' [-Wunused-variable]
/home/mike/learn...	307	warning: unused variable 'inloop' [-Wunused-variable]
/home/mike/learn...	308	warning: unused variable 'setF' [-Wunused-variable]
/home/mike/learn...	187	warning: variable 'num_hours' set but not used [-Wunused-but-set-variable]
/home/mike/learn...	330	warning: 'myDist' may be used uninitialized in this function [-Wmaybe-uninitialized]
=== Build finished: 0 error(s), 13 warning(s) (0 minute(s), 8 second(s)) ===		

- In the 'build' folder: 'sudo make install', 'sudo make package', using 'Terminal'.

```
mike@mike-TECRA-A10: ~/learning/oplaydo1_pi/build
CPack: Create package using DEB
CPack: Install projects
CPack: - Run preinstall target for: oplaydo1_pi
CPack: - Install project: oplaydo1_pi
CPack: Create package
CPack: - package: /home/mike/learning/oplaydo1_pi/build/oplaydo1_pi_0.1-1_i386.deb generated.
CPack: Create package using RPM
CPack: Install projects
CPack: - Run preinstall target for: oplaydo1_pi
CPack: - Install project: oplaydo1_pi
CPack: Create package
CPackRPM: Will use GENERATED spec file: /home/mike/learning/oplaydo1_pi/build/_CPack_Packages/Linux/RPM/SPECS/oplaydo1_pi.spec
CPack: - package: /home/mike/learning/oplaydo1_pi/build/oplaydo1_pi_0.1-1_i386.rpm generated.
CPack: Create package using TBZ2
CPack: Install projects
CPack: - Run preinstall target for: oplaydo1_pi
CPack: - Install project: oplaydo1_pi
CPack: Create package
CPack: - package: /home/mike/learning/oplaydo1_pi/build/oplaydo1_pi_0.1-1_i386.tar.bz2 generated.
mike@mike-TECRA-A10:~/learning/oplaydo1_pi/build$
```

- This creates a '.deb' package which can be opened with the 'Ubuntu Software Centre'.

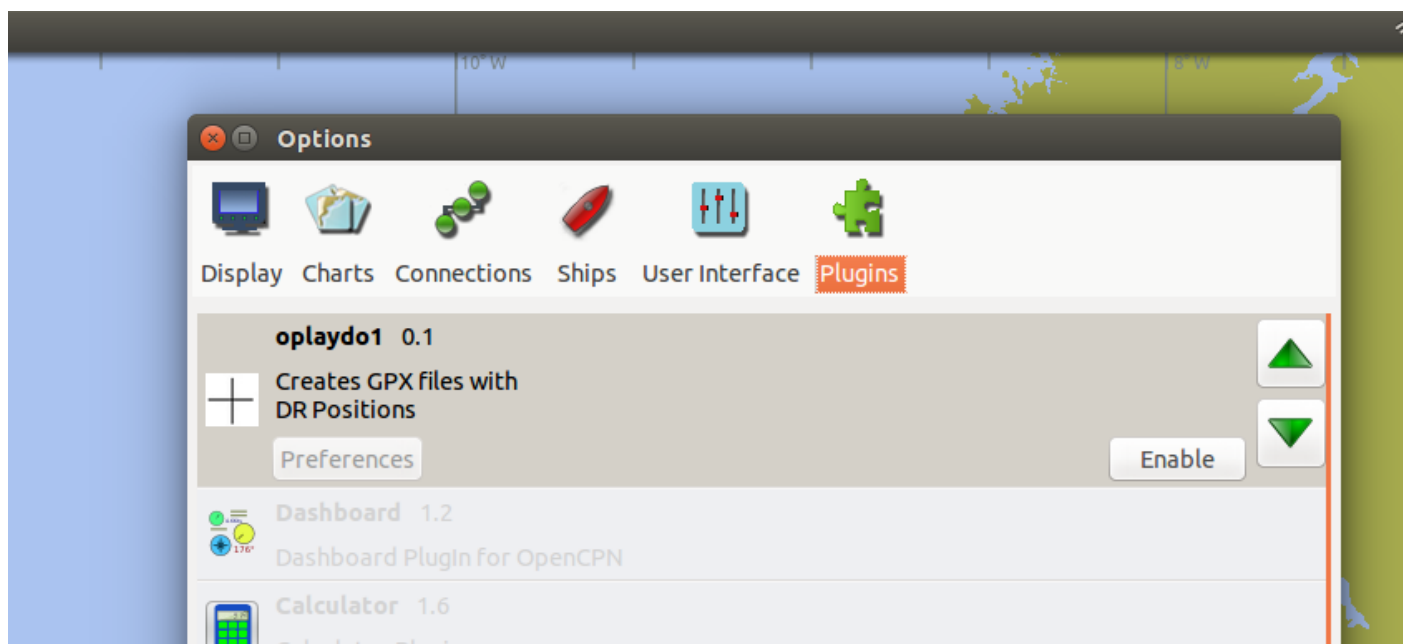


- Using the 'Ubuntu Software Centre' 'oplaydo_pi' can be installed, in the same way as 'DR_pi'.

- On the first install the plugin was still named 'DR_pi' not 'oplaydo1_pi'. There is code in 'oplaydo1_pi.cpp' that needs editing with CodeBlocks. Change 'DR' to 'oplaydo1'. '_img_DR' was changed to '_img_oplaydo1' in 'oplaydo1_pi.cpp' and in 'icons.cpp' and 'icons.h'.

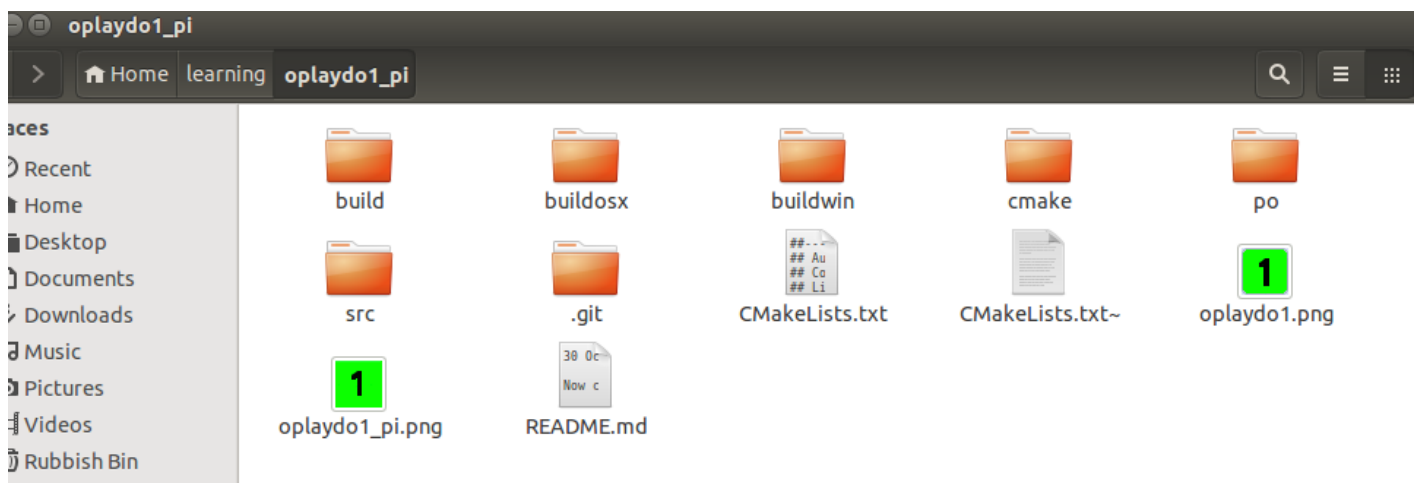
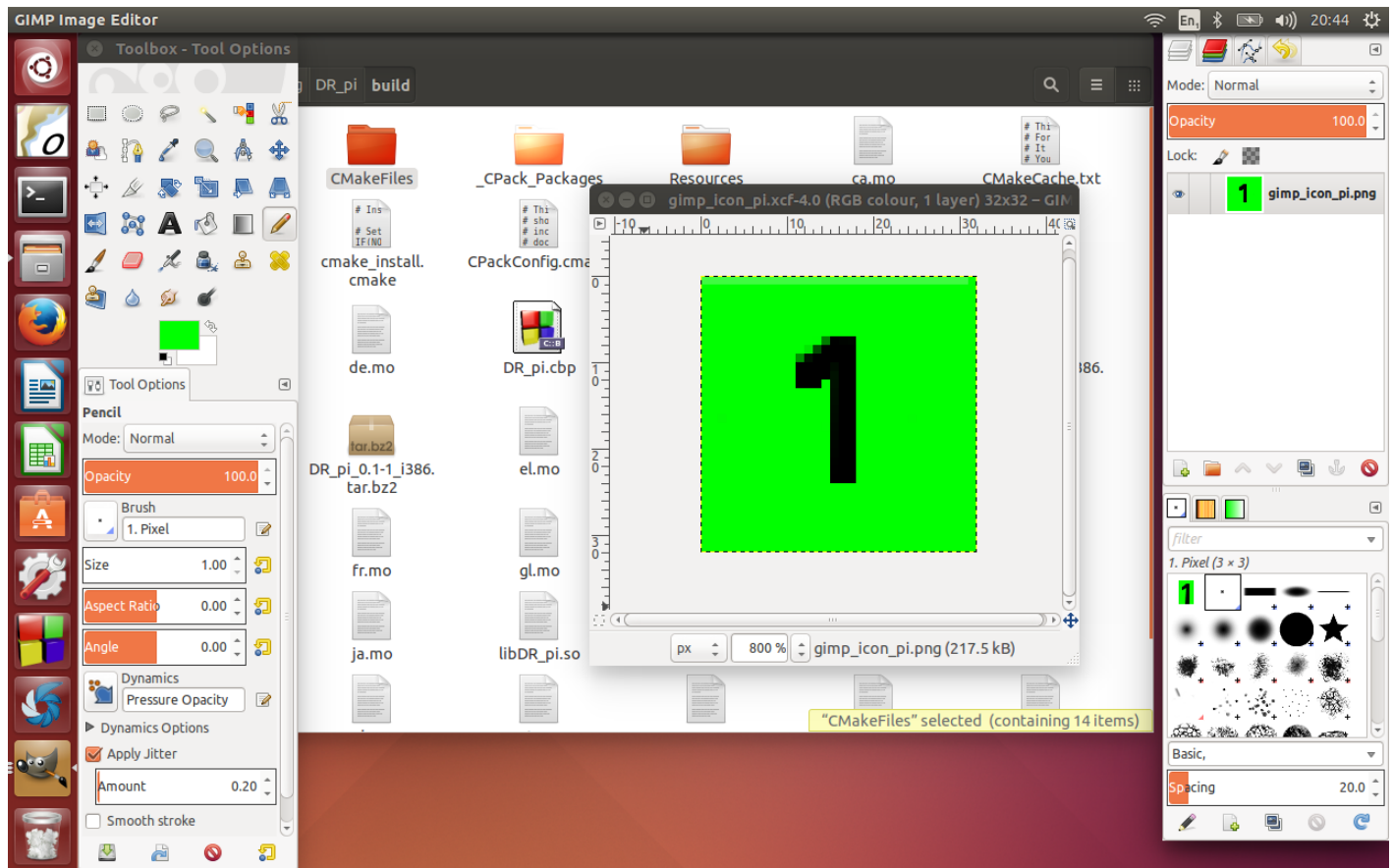
- Then 'build' again, followed by 'sudo make install', 'sudo make package', using 'Terminal'.

- The icon has not yet been changed, or any of the code to customise our new plugin.



Using GIMP to make new icons

- Using [Shutter](#) a screenshot was made from the 'DR_pi' icons as a starting point for the new icons.
- The '.png' files were then opened in Gimp. - Size of the images was adjusted to 32×32 and the 'Mode' was set as 'indexed' with 16 colours.
- The images were then **exported** as new '.png' files and saved in the '/learning/oplaydo1_pi' folder, as 'oplaydo1.png' and 'oplaydo1_pi.png'.



Using png2wx.pl

- The two icon images that were created must now be translated into a form that can be used by wxWidgets.
- Using 'Ubuntu Software Centre' check that 'Perl' has been installed on your machine.
- Download the script 'png2wx.pl' from [here](#). Place the script in the 'oplaydo1_pi' root folder.
- Using 'Terminal' navigate to 'learning/oplaydo1_pi'. Type the following command line.

```
perl png2wx.pl -C icons.cpp -H icons.h -M ICONS_H oplaydo1.png oplaydo1_pi.png
```

- Two files, 'icons.cpp' and 'icons.h' are created in the root folder.
- Copy these files to the 'learning/oplaydo1_pi/src' folder. They will replace the existing files and provide the new icons for 'oplaydo1_pi'.
- Using 'CodeBlocks' open the 'oplaydo1_pi' project.
- Build the project again.
- Using 'Terminal' cd to the 'build' folder. 'sudo make install', 'sudo make package'.
- The '.deb' package made can be opened with the 'Ubuntu Software Centre' and reinstalled.



- To summarise. The plugin has been renamed and the icons changed to match the new project. At present the functionality in 'oplaydo1_pi' is nearly the same as that in 'DR_pi'.
- In the Part 2 the plugin is given new functionality.

Part 2: Edit the plugin for new user actions

Additional requirements

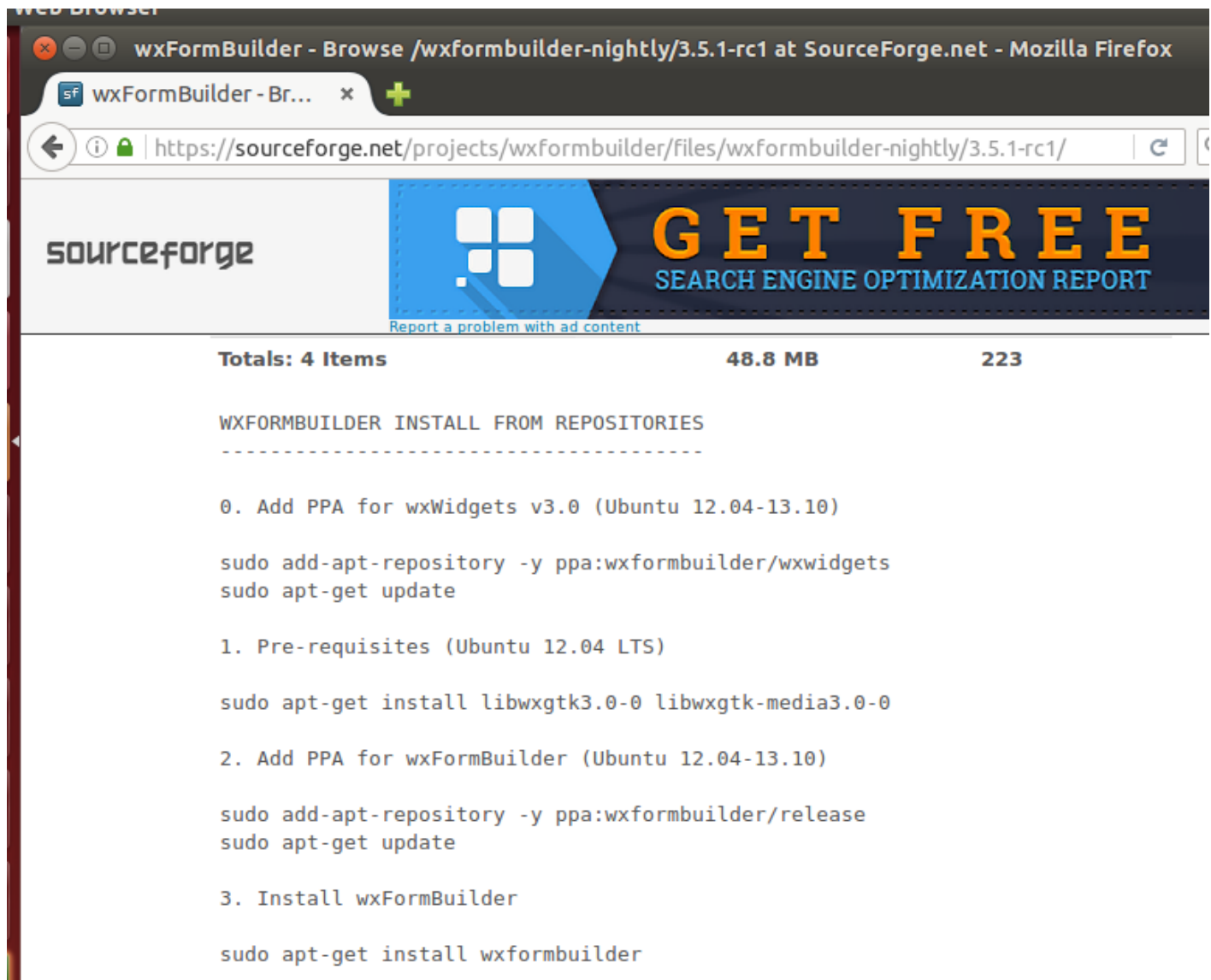
- Using the 'Ubuntu Software Centre' install '[wxFormBuilder](#)'. (RAD tool for wxWidgets GUI design)

Aim

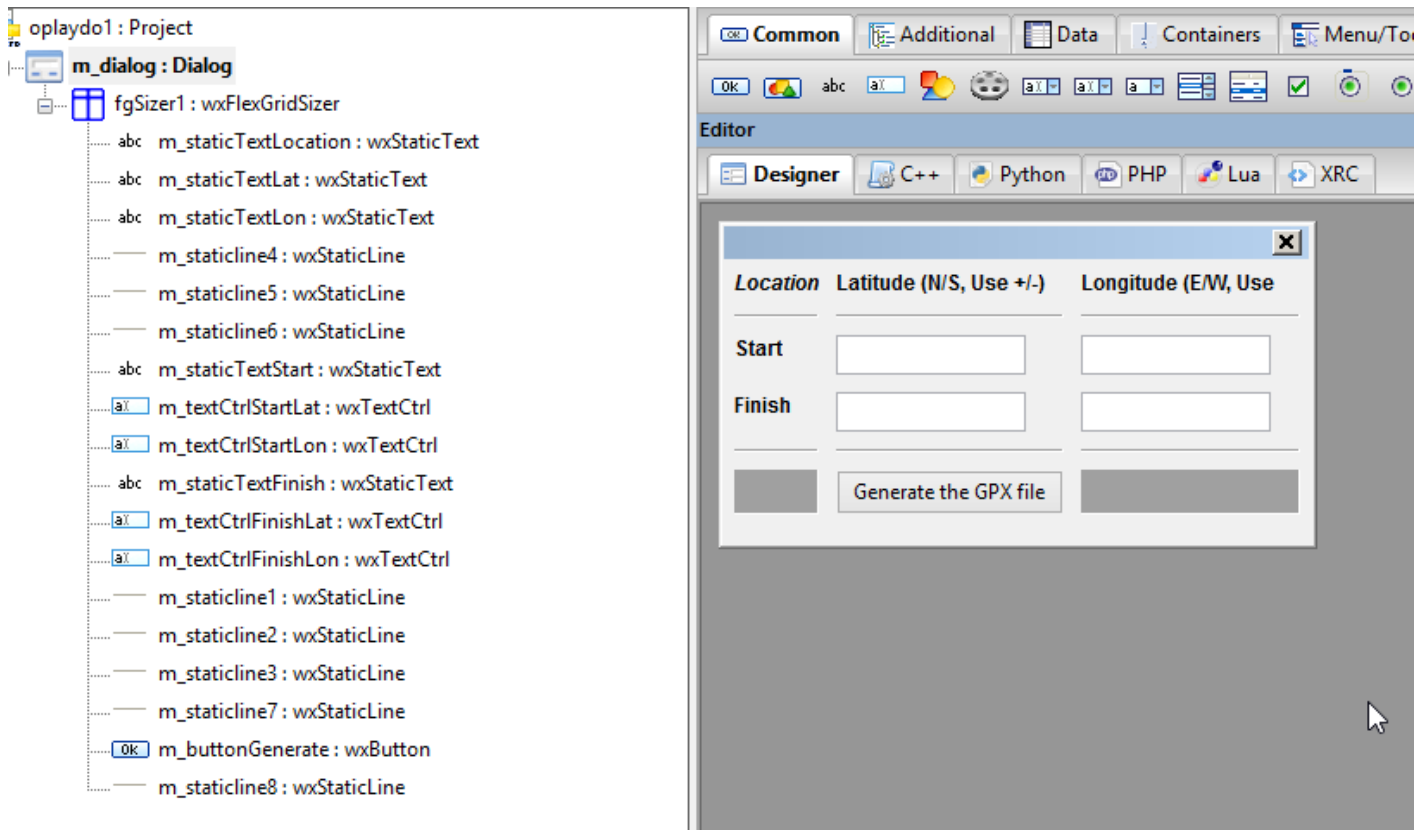
- Make a new user interface and generate the code for this.
- Modify the plugin code to use the new interface
- Remove unwanted code

User Interface

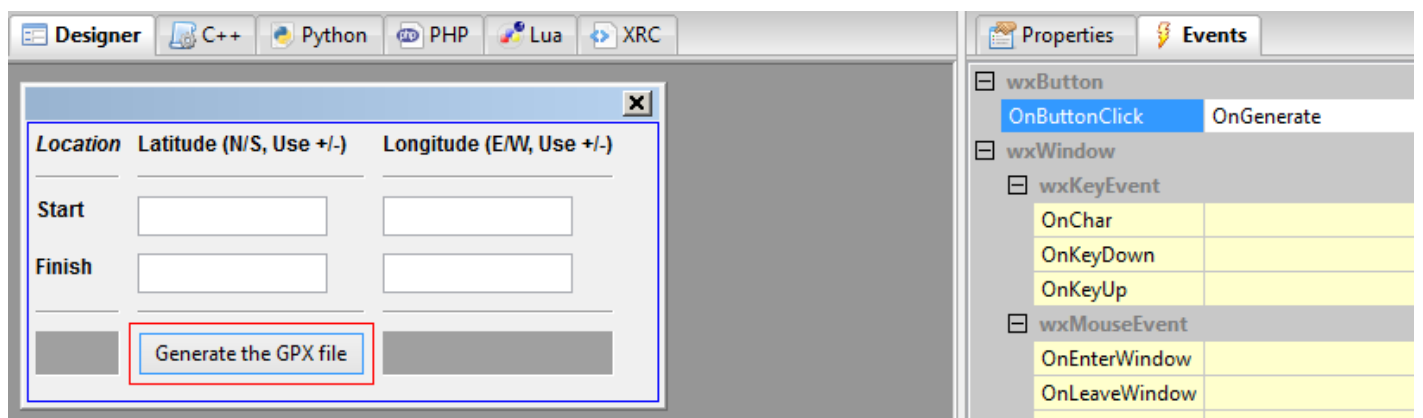
- To save time the 'oplaydo1.fbp' wxFormBuilder project file was copied from the Windows machine.
- **Problem** 'Ubuntu Software Centre' had installed a copy of wxFormBuilder that did not open the '.fbp'. Although the wxFormBuilder source etc has moved to GitHub, SourceForge still has some useful information. Following these instructions wxFormBuilder was updated, even though my Linux machine is using 'Ubuntu 14.04 - Trusty'.



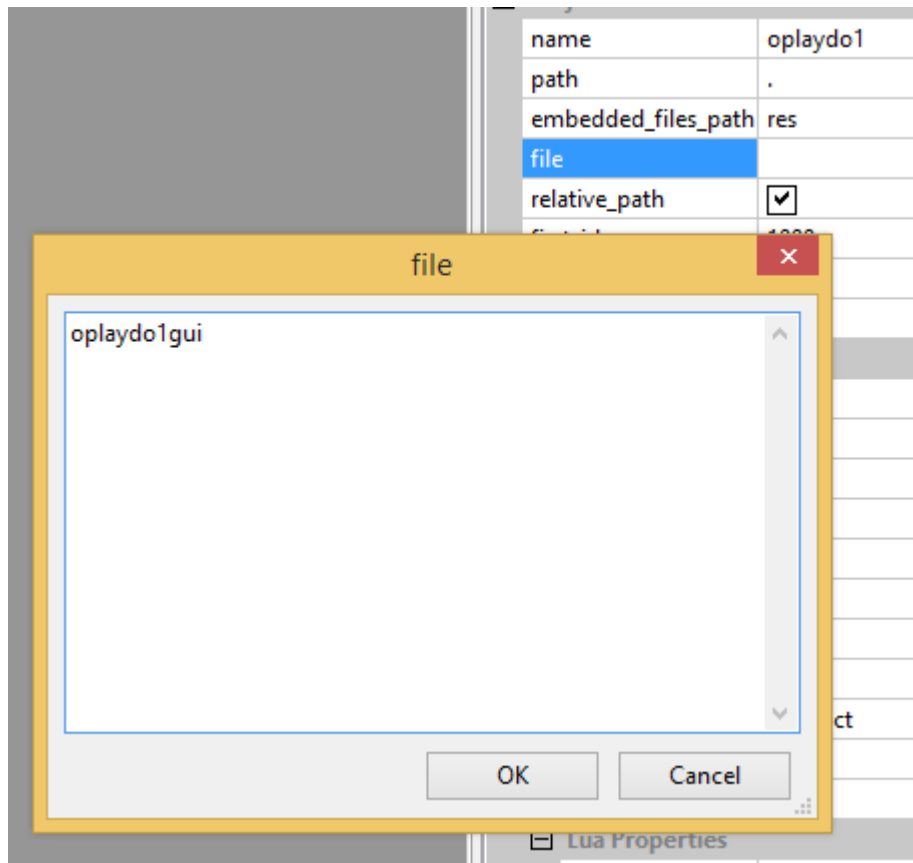
- The 'oplaydo1.fbp' file could now be opened.
- Alternatively start a new project in wxFormBuilder. Edit the properties. The project will be called 'oplaydo1'.
- This is what you are aiming for:



- The textboxes (textCtrl) have been given names that indicate their use.
- An event called 'OnGenerate' has been added to the wxButton 'm_buttonGenerate'. When the plugin is used this button will generate the GPX file. The format can be imported into OpenCPN.



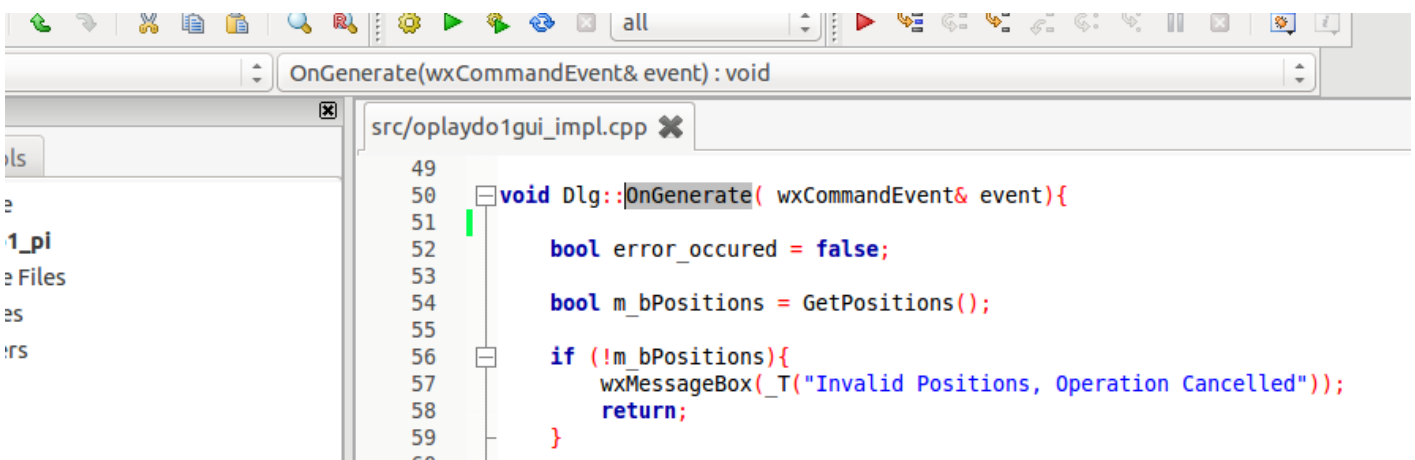
- In the project properties set 'file' as 'oplaydo1gui'. The path in my case is 'home/mike/learning/oplaydo1_pi'.



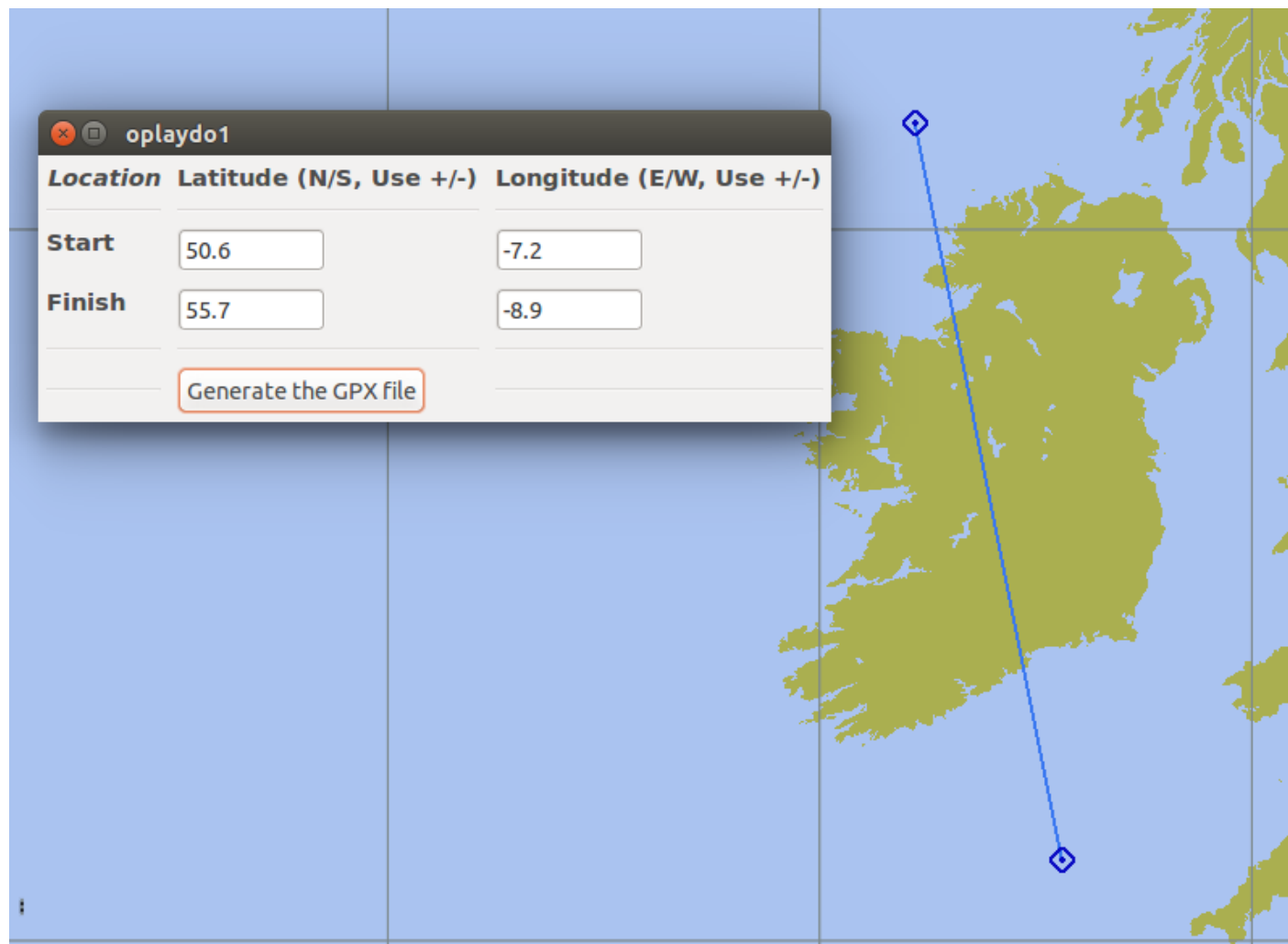
- When you are satisfied with the design use 'File/Generate Code' (or F8) to make two files in the root folder, 'oplaydo1gui.cpp' and 'oplaydo1gui.h'.

Modifying the code

- Make a backup of the files 'oplaydo1gui.cpp' and 'oplaydo1gui.h' in 'src' in case you need to refer to them.
- Move 'oplaydo1gui.cpp' and 'oplaydo1gui.h' from 'oplaydo1_pi' to 'oplaydo1_pi/src'.
- Open the 'oplaydo1_pi' project with 'CodeBlocks'.
- 'DlgDef' needs changing to 'm_dialog', which was the class wxFormBuilder used for the dialog.
- 'DlgCfgDef' sections are removed. We will not be using 'preferences' for the plugin.
- In 'oplaydo1gui_impl.h' change 'DlgDef' to 'm_dialog'.
- Try to avoid any changes to 'oplaydo1gui.cpp' and 'oplaydo1gui.h'. This would create differences with the source wxFormBuilder file.
- The command event 'OnGenerate' has to be linked to a function in 'oplaydo1gui_impl.h'.



- The function that has the code we want for generating a GPX file is originally called 'Calculate'. Change 'Calculate' to 'OnGenerate'.
- In 'oplaydo1gui_impl.cpp' find the function 'Calculate' and rename it to 'OnGenerate'.
- The functions 'OnPSGPX' is not needed. Parts of 'OpenXML' may be useful. Either delete functions/code or comment out.
- 'Preferences' is not going to be used so 'CfgDlg' references are removed.
- All the changes are not going to be listed. Source code files can be downloaded [here](#).
- 'std::abs' causes an error on Linux, which is found by CodeBlocks. Use 'fabs' instead.
- **Problem ...**After 'sudo make install', 'sudo make package' and installing the '.deb' package no icons could be seen in OpenCPN.
- **Remedy ...**One of the '.png' icon images still used 256 colours. Using GIMP this was 'mode/indexed' to 16. The png2wx.pl script was rerun and the two files 'icons.cpp' and 'icons.h' copied to the 'src' folder. The project was rebuilt in CodeBlocks.
- 'sudo make install', 'sudo make package'. Reinstall the '.deb' package.
- After removing all the errors and building/installing the plugin the new dialog form is shown and tested.



From:
<https://opencpn.org/wiki/dokuwiki/> - OpenCPN Manuals

Permanent link:
https://opencpn.org/wiki/dokuwiki/doku.php?id=opencpn:developer_manual:plugins:learning:oplaydo1_linux

Last update: 2017/02/21 17:52

